

CLAIMS

What is claimed is:

Sub
A1

- 1 1. An apparatus comprising:
 - 2 a processor cache unit to process a cache access request from a
 - 3 processor core of a processor, the processor cache unit including a
 - 4 processor cache controller and a processor cache; and
 - 5 a chipset cache controller coupled to the processor cache unit to
 - 6 control a chipset cache located in a chipset in response to the cache
 - 7 access request from the processor core, the chipset being coupled to
 - 8 the processor via a bus.
- 1 2. The apparatus of claim 1 wherein the chipset cache
 - 2 controller comprises:
 - 3 a chipset cache tag store to store tags corresponding to cache
 - 4 lines of the chipset cache; and
 - 5 a coherency controller coupled to the chipset cache tag store to
 - 6 maintain cache coherency among the processor cache, the chipset
 - 7 cache, and a memory, according to a coherence protocol.
- 1 3. The apparatus of claim 2 wherein the coherency protocol is
 - 2 a modified, exclusive, share, and invalidated (MESI) protocol.

1 4. The apparatus of claim 3 wherein the coherence controller
2 comprises:

3 a chipset interface circuit to send control signals to the chipset
4 according to cache state and type of the cache access request, the
5 control signals specifying an operation performed by the chipset.

1 5. The apparatus of claim 4 wherein the control signals
2 include at least a set identifier for a cache set in the chipset cache
3 corresponding to the cache access request, a cache valid indicator
4 asserted when a cache line in the cache set is valid, and a flush
5 indicator asserted when the cache line is flushed.

1 6. The apparatus of claim 5 wherein when the type of the
2 cache access request is a read request and the cache valid indicator is
3 not asserted, the operation includes one of a transfer of a data read
4 from the memory to the cache set in the chipset cache and a transfer of
5 a data read from the memory to the processor.

1 7. The apparatus of claim 6 wherein when the flush indicator
2 is asserted, the operation further includes a flushing of existing data at
3 the cache set.

1 8. The apparatus of claim 4 wherein when the type of the
2 cache access request is a read request and the cache valid indicator is

3 asserted, the operation includes a transfer of a data read from the
4 cache set to the processor.

Such
AI/

1 9. The apparatus of claim 4 wherein when the type of the
2 cache access request is a write request, the operation includes a
3 transfer of a data from the processor to the cache set in the chipset
4 cache.

1 10. The apparatus of claim 9 wherein when the cache valid
2 indicator is not asserted, the operation further includes a transfer of the
3 data from the processor to the memory.

1 11. The apparatus of claim 9 wherein when the flush indicator
2 is asserted, the operation further includes a flushing of existing data at
3 the cache set.

1 12. The apparatus of claim 5 further comprising:
2 a snoop circuit coupled to the chipset cache tag store to check if
3 an address snooped on the bus matches with one of entries in the
4 chipset cache tag store.

1 13. The apparatus of claim 12 wherein the set identifier
2 specifies the cache set corresponding to the one of the entries that
3 matches the address snooped on the bus.

Sub
A17

1 14. A method comprising:
2 processing a cache access request from a processor core of a
3 processor by a processor cache unit, the processor cache unit including
4 a processor cache controller and a processor cache; and
5 controlling a chipset cache located in a chipset in response to the
6 cache access request from the processor core, the chipset being
7 coupled to the processor via a bus.

1 15. The method of claim 14 wherein controlling the chipset
2 cache comprises:
3 storing tags corresponding to cache lines of the chipset cache in
4 a chipset cache tag store; and
5 maintaining cache coherency among the processor cache, the
6 chipset cache, and a memory, according to a coherence protocol.

1 16. The method of claim 15 wherein the coherency protocol is
2 a modified, exclusive, share, and invalidated (MESI) protocol.

1 17. The method of claim 16 wherein maintaining cache
2 coherency comprises:

3 sending control signals to the chipset according to cache state
4 and type of the cache access request, the control signals specifying an
5 operation performed by the chipset.

1 18. The method of claim 17 wherein the control signals include
2 at least a set identifier for a cache set in the chipset cache
3 corresponding to the cache access request, a cache valid indicator
4 asserted when a cache line in the cache set is valid, and a flush
5 indicator asserted when the cache line is flushed.

1 19. The method of claim 18 wherein when the type of the cache
2 access request is a read request and the cache valid indicator is not
3 asserted, the operation includes one of a transfer of a data read from
4 the memory to the cache set in the chipset cache and a transfer of a
5 data read from the memory to the processor.

1 20. The method of claim 19 wherein when the flush indicator is
2 asserted, the operation further includes a flushing of existing data at
3 the cache set.

1 21. The method of claim 17 wherein when the type of the cache
2 access request is a read request and the cache valid indicator is
3 asserted, the operation includes a transfer of a data read from the
4 cache set to the processor.

Sub
A17

1 22. The method of claim 17 wherein when the type of the cache
2 access request is a write request, the operation includes a transfer of a
3 data from the processor to the cache set in the chipset cache.

1 23. The method of claim 22 wherein when the cache valid
2 indicator is not asserted, the operation further includes a transfer of the
3 data from the processor to the memory.

1 24. The method of claim 22 wherein when the flush indicator is
2 asserted, the operation further includes a flushing of existing data at
3 the cache set.

1 25. The method of claim 18 further comprising:
2 checking if an address snooped on the bus matches with one of
3 entries in the chipset cache tag store.

1 26. The method of claim 25 wherein the set identifier specifies
2 the cache set corresponding to the one of the entries that matches the
3 address snooped on the bus.

1 27. A system comprising:
2 a memory to store data;
3 a chipset coupled to memory having a chipset cache; and

Sub
A1

4 a processor coupled to the memory and the chipset via a bus, the
5 processor including a processor core and a cache unit, the cache unit
6 comprising:

7 a processor cache unit to process a cache access request
8 from the processor core, the processor cache unit including
9 a processor cache controller and a processor cache, and
10 a chipset cache controller coupled to the processor cache
11 unit to control the chipset cache in response to the cache
12 access request from the processor core.

1 28. The system of claim 27 wherein the chipset cache controller
2 comprises:

3 a chipset cache tag store to store tags corresponding to cache
4 lines of the chipset cache; and

5 a coherency controller coupled to the chipset cache tag store to
6 maintain cache coherency among the processor cache, the chipset
7 cache, and a memory, according to a coherence protocol.

1 29. The system of claim 28 wherein the coherency protocol is a
2 modified, exclusive, share, and invalidated (MESI) protocol.

1 30. The system of claim 29 wherein the coherence controller
2 comprises:

Sub
Al

3 a chipset interface circuit to send control signals to the chipset
4 according to cache state and type of the cache access request, the
5 control signals specifying an operation performed by the chipset.

1 31. The system of claim 30 wherein the control signals include
2 at least a set identifier for a cache set in the chipset cache
3 corresponding to the cache access request, a cache valid indicator
4 asserted when a cache line in the cache set is valid, and a flush
5 indicator asserted when the cache line is flushed.

1 32. The system of claim 31 wherein the cache unit further
2 comprising

3 a snoop circuit coupled to the chipset cache tag store to check if
4 an address snooped on the bus matches with one of entries in the
5 chipset cache tag store.